

Научная статья

УДК 004.896

<https://doi.org/10.37493/2307-910X.2025.2.3>

## Применение нейронных сетей для автоматизации тестирования пользовательского интерфейса

**Игорь Георгиевич Волынец<sup>1\*</sup>**<sup>1</sup> Юзтех Бел (д. 3Б, пр-т Дзержинского, Минск, 220069, Беларусь)<sup>1</sup> [ivolynets@usetech.ru](mailto:ivolynets@usetech.ru)

\*Автор, ответственный за переписку

**Аннотация.** В статье рассматривается применение нейронных сетей для автоматизации тестирования пользовательского интерфейса (UI). Нейронные сети, благодаря их способности к обучению и адаптации, значительно повышают эффективность и точность процессов тестирования. В работе освещены различные подходы к применению нейросетевых технологий в автоматизации UI-тестирования, включая генерацию и оптимизацию тестовых сценариев, классификацию и анализ дефектов, а также прогнозирование результатов тестирования. Особое внимание уделено преимуществам, которые дают нейронные сети в улучшении процессов тестирования пользовательского интерфейса, сокращении времени разработки и увеличении качества конечного продукта. Однако, несмотря на очевидные выгоды, внедрение этих технологий связано с рядом вызовов, требующих дополнительных исследований и ресурсов. Статья подчеркивает важность интеграции нейронных сетей в современные процессы разработки для достижения высоких стандартов качества программного обеспечения.

**Ключевые слова:** нейронные сети, автоматизация тестирования, пользовательский интерфейс, UI-тестирование, искусственный интеллект, качество программного обеспечения, генерация тестов

**Для цитирования:** Волынец И. Г. Применение нейронных сетей для автоматизации тестирования пользовательского интерфейса // Современная наука и инновации. 2025. № 2. С. 32–40. <https://doi.org/10.37493/2307-910X.2025.2.3>

Research article

## The use of neural networks to automate user interface testing

**Igor G. Volynets<sup>1\*</sup>**<sup>1</sup> Usetech Bel (3B, Dzerzhinsky Ave., Minsk, 220069, Belarus)<sup>1</sup> [ivolynets@usetech.ru](mailto:ivolynets@usetech.ru)

\*Corresponding author

**Abstract.** The article discusses the use of neural networks to automate user interface (UI) testing. Neural networks, due to their ability to learn and adapt, significantly increase the efficiency and accuracy of testing processes. The paper highlights various approaches to the application of neural network technologies in the automation of UI testing, including the generation and optimization of test scenarios, classification and analysis of defects, as well as forecasting test results. Special attention is paid to the advantages that neural networks provide in improving user interface testing processes, reducing development time and increasing the quality of the final product. However, despite the obvious benefits, the introduction of these technologies is associated with a number of challenges that require additional research and resources. The article emphasizes the importance of integrating neural networks into modern development processes to achieve high software quality standards.

© Волынец И. Г., 2025

**Keywords:** neural networks, test automation, user interface, UI testing, artificial intelligence, CI/CD, software quality, test generation

**For citation:** Volynets IG. The use of neural networks to automate user interface testing. *Modern Science and Innovations*. 2025;(2):32-40. (In Russ.). <https://doi.org/10.37493/2307-910X.2025.2.3>

**Введение.** В последние годы стремительное развитие технологий искусственного интеллекта и, в частности, нейронных сетей оказало значительное влияние на различные области деятельности человека. Одним из ключевых направлений, где нейронные сети демонстрируют свою эффективность, является автоматизация процессов тестирования программного обеспечения, особенно пользовательских интерфейсов (UI). Тестирование UI играет важнейшую роль в обеспечении качества программ, именно взаимодействие с интерфейсом определяет пользовательский опыт и восприятие продукта в целом.

Актуальность темы обусловлена возрастающей сложностью современных программных продуктов и необходимостью быстрого и точного тестирования интерфейсов в условиях жесткой конкуренции на рынке цифровых технологий. В этом контексте использование нейронных сетей для автоматизации тестирования UI становится перспективным решением, способным не только снизить затраты на тестирование, но и повысить его качество.

Целью данной работы является исследование возможностей применения нейронных сетей для автоматизации тестирования пользовательского интерфейса.

**Основы нейронных сетей и их применение в тестировании UI.** Применение нейронных сетей в тестировании пользовательского интерфейса (UI) открывает новые возможности для повышения качества и эффективности процессов тестирования. В силу того, что традиционные методы тестирования UI часто требуют значительных временных и человеческих ресурсов, что делает процесс тестирования длительным и трудоемким. Использование нейронных сетей позволяет автоматизировать процессы тестирования UI, что ведет к снижению затрат и увеличению скорости разработки [1]. Например, нейронные сети могут использоваться для автоматического распознавания визуальных элементов интерфейса, таких как кнопки, поля ввода и иконки. Это помогает выявлять несоответствия в дизайне или функциональности интерфейса на ранних стадиях разработки. Кроме того, нейронные сети способны анализировать поведение пользователей при взаимодействии с интерфейсом, что позволяет выявить потенциальные проблемы в удобстве использования и адаптировать интерфейс под нужды конечных пользователей [2]. Сети могут состоять из слоев, которые будут расписаны в таблице 1.

**Таблица 1. Слои нейронных сетей [2] /**  
**Table 1. Layers of neural networks [2]**

| Наименование слоя | Описание                                                                                                                                                                                                                                                                                                                                                   |
|-------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Входной слой      | Этот слой отвечает за получение начальных данных, представляющих объяснительные переменные для каждого наблюдения. Количество узлов во входном слое обычно соответствует числу этих переменных. Функция входного слоя заключается в передаче данных на последующие слои без изменений, что позволяет им быть обработанными на более глубоких уровнях сети. |
| Скрытые слои      | В скрытых слоях происходит основная обработка данных. Входные сигналы, поступающие из входного слоя или других скрытых слоев, подвергаются математическим преобразованиям с использованием весовых коэффициентов, заложенных в сети. Эти преобразования позволяют выявить сложные зависимости и паттерны в данных, что повышает точность прогнозов.        |
| Выходной слой     | Завершающим элементом нейронной сети является выходной слой, который принимает обработанные данные от скрытых слоев и формирует окончательный результат. В задачах классификации, как правило, используется один выходной узел, обеспечивающий предсказание значения целевой переменной.                                                                   |

Структура нейронной сети, известная как архитектура или топология, определяется количеством слоев, числом узлов в каждом слое и механизмом регулировки весовых коэффициентов. Наиболее простой конфигурацией считается двухслойная сеть, состоящая из входного и выходного слоев, где каждый узел связан с соответствующим входом и выходом, обеспечивая минимальную, но достаточную функциональность для простых задач.

Для лучшего понимания работы искусственного нейрона и преобразования входных сигналов можно рассмотреть его внутреннюю структуру и механизмы обработки данных. Полная модель искусственного нейрона представлена на рисунке 1.

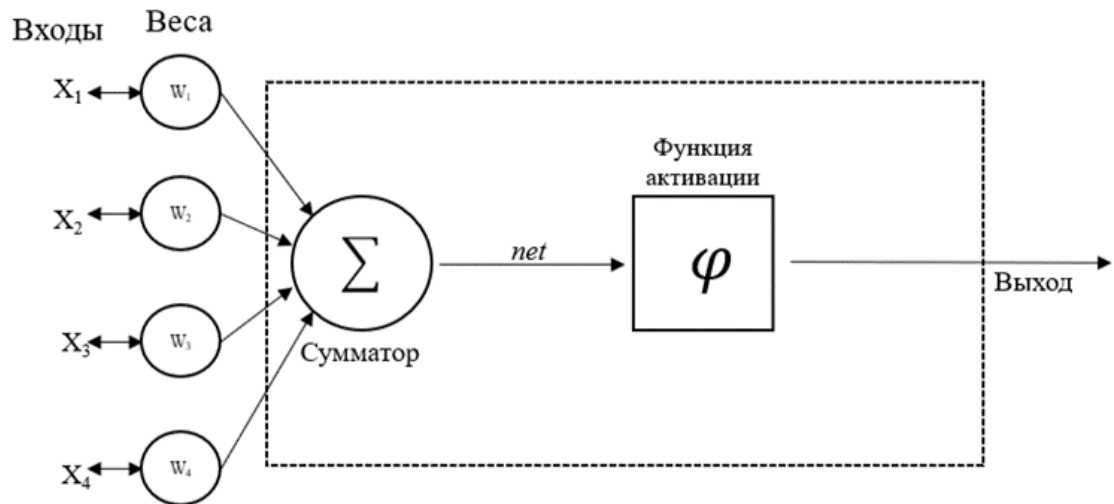


Рисунок 1. Внутренняя структура искусственного нейрона [3] /  
Figure 1. Internal structure of an artificial neuron [3]

В математических терминах нейрон срабатывает тогда и только тогда, когда

$$X_1 \times W_1 + X_2 \times W_2 + X_3 \times W_3 \dots > T(1),$$

где:

$X_1, X_2, X_3, \dots$  – это входные сигналы или значения, поступающие на нейрон. Например, это могут быть данные, которые нейрон получает на вход, такие как значения пикселей изображения или характеристики объекта.

$W_1, W_2, W_3, \dots$  – это веса (коэффициенты) нейрона, которые определяют, насколько важно каждое входное значение для принятия решения нейроном. Веса являются настраиваемыми параметрами модели, которые обучаются в процессе тренировки нейронной сети.

$T$  – это пороговое значение (threshold), которое определяет, при каком суммарном взвешенном значении входных сигналов нейрон "срабатывает". Это фиксированное значение, которое сравнивается с суммой входных сигналов, умноженных на соответствующие веса.

$\Sigma$  (суммирование) – вся конструкция  $X_1 \times W_1 + X_2 \times W_2 + X_3 \times W_3 + \dots$  представляет собой взвешенную сумму входных сигналов.

Добавление весовых коэффициентов для входов и порогового значения делает данный нейрон весьма универсальным и мощным инструментом. Модель нейрона МакКаллока-Питтса отличается способностью приспосабливаться к конкретным условиям, путем изменения своих весовых коэффициентов или порога активации[3].

Далее рассмотрим виды нейронных сетей, использующиеся в процессе автоматизации тестирования пользовательского интерфейса. Простейшие нейронные сети, представленные однослойными персептронами, включают в себя лишь один слой

нейронов, который получает входные сигналы и генерирует выходные значения. Эти сети являются базовой формой нейронных сетей и часто применяются для решения задач классификации, распознавания образов и прогнозирования.

Многослойные персептроны (МЛП) состоят из нескольких слоев, включая входной, скрытые слои и выходной слой. Взаимодействие между этими слоями обеспечивает сложную обработку данных. МЛП находят применение в решении сложных задач, таких как распознавание речи, анализ текста, предсказание временных рядов и другие аналогичные задачи.

Рекуррентные нейронные сети (РНС) отличаются наличием циклических связей, которые позволяют использовать результаты предыдущих шагов обработки в качестве входных данных для последующих этапов. Эта особенность делает их незаменимыми в задачах, связанных с временными рядами, последовательной обработкой данных и анализом текста, где контекст играет важную роль.

Сверточные нейронные сети (СНС) специально разработаны для обработки визуальных данных, таких как изображения и видео. Они используют свертку для выделения значимых признаков из входных данных и пулинг, выступающий в роли механизма обобщения низкоуровневых данных, позволяющий нейронной сети переходить от данных с высоким разрешением до информации с более низким разрешением.

Глубокие нейронные сети (ГНС) состоят из множества слоев (обычно более трех) и предназначены для решения сложных задач, таких как распознавание изображений, обработка естественного языка, обучение с подкреплением и генерация контента. ГНС обладают высокой гибкостью и способностью автоматически извлекать высокоуровневые признаки из сложных иерархических данных [4].

Таким образом, благодаря использованию разных видов нейронных сетей в процессе тестирования пользовательского интерфейса, удастся добиться повышения эффективности и точности проверок интерфейсов. Что достигается за счет возможностей, которыми обладают нейронные сети, выраженные в анализе визуальных элементов, распознавания текста и оценивании взаимодействия пользователя с интерфейсом, что обеспечивает более глубокий уровень понимания и выявления ошибок, чем традиционные методы. Например, с помощью сверточных нейронных сетей можно автоматизировать процесс распознавания изображений, определяя, корректно ли отображаются графические элементы на экране. Кроме того, рекуррентные нейронные сети могут применяться для анализа последовательности действий пользователя, помогая выявить возможные сбои в сценариях использования.

Далее рассмотрим общетеоретические аспекты UX/UI тестирования, которое представляет собой важнейший методологический подход, который позволяет дизайнерам понять, как пользователи взаимодействуют с цифровыми продуктами. В действительности, это не просто отдельный инструмент, а целый комплекс методик, нацеленных на глубокое изучение пользовательского опыта. Опросы и интервью, в частности, играют ключевую роль в выявлении предпочтений и потребностей пользователей, что позволяет получить ценные инсайты для дальнейшего улучшения продукта.

Проведение UX/UI исследований приносит существенную пользу бизнесу, позволяя не только убедиться в актуальности продукта для целевой аудитории, но и получить объективную обратную связь от пользователей. Например, глубинные интервью помогают выявить основные задачи и проблемы, которые пользователи пытаются решить с помощью продукта. В то же время, опросы позволяют оценить удобство использования приложения в сравнении с конкурентами, а также определить, какие новые функции могут повысить ценность продукта для конечных пользователей [5].

Применение нейронных сетей в тестировании пользовательского интерфейса (UI) представляет собой перспективное направление, способное значительно повысить эффективность и точность процесса тестирования. Традиционные методы тестирования

UI требуют значительных временных и людских ресурсов, что особенно актуально в условиях частых обновлений и изменений интерфейса. Нейронные сети, благодаря своей способности к обучению на больших объемах данных и выявлению сложных закономерностей, могут автоматизировать многие аспекты тестирования и сделать этот процесс более адаптивным.

Нейросети могут обучаться распознаванию различных компонентов UI, таких как кнопки, текстовые поля и иконки, и сравнивать их с эталонными образцами. Это позволяет автоматически выявлять отклонения от заданного дизайна или функционала, которые могут возникать при внесении изменений в код или дизайне.

Кроме того, нейронные сети могут быть использованы для моделирования поведения пользователей. Обученные на данных о реальных взаимодействиях с интерфейсом, они способны предсказывать потенциальные ошибки пользователей и выявлять элементы, вызывающие затруднения. Это позволяет не только автоматизировать тестирование, но и повысить качество пользовательского опыта за счет выявления проблем на ранних стадиях разработки. В свою очередь некоторые компании уже активно используют нейронные сети для автоматизации тестирования пользовательских интерфейсов (UI), к таким компаниям относятся:

1. Google: Google применяет нейронные сети в рамках тестирования своих приложений и сервисов, таких как Google Chrome, а также ОС Android. Например, Google использует нейронные сети для автоматического создания и оптимизации тестов, что позволяет ускорить процесс тестирования и повысить его качество. Это включает в себя анализ изменений в коде и интерфейсе для автоматического обновления тестовых сценариев [6].

2. Microsoft: В Microsoft активно используют нейронные сети для тестирования продуктов, таких как Microsoft Office и Windows. Инструменты на основе нейронных сетей помогают автоматизировать тестирование пользовательского интерфейса, снижая количество ручной работы и повышая точность обнаружения дефектов. Например, Microsoft использует Playwright, который поддерживает кросс-браузерное тестирование и тестирование на различных устройствах с использованием нейронных сетей [7].

3. Facebook (Meta): Facebook также применяет нейронные сети для автоматизации тестирования своих приложений, включая Facebook и Instagram. Нейронные сети используются для анализа пользовательских взаимодействий и автоматического создания тестов, что позволяет компании обеспечивать высокое качество своих продуктов в условиях постоянных обновлений и изменений интерфейса [7].

Таким образом, на основании опыта данных компаний можно заметить, что использование нейронных сетей в автоматизации тестирования пользовательского интерфейса, позволяет обеспечивать высокое качество продуктов и сокращать время на тестирование.

**Преимущества и вызовы использования нейронных сетей для автоматизации UI-тестирования.** Одним из основных преимуществ использования нейронных сетей в автоматизации тестирования пользовательского интерфейса является высокая степень автоматизации процесса. Традиционные методы требуют значительного участия человека для создания тестовых сценариев и анализа результатов. Нейронные сети, в свою очередь, могут автоматически генерировать тестовые сценарии, основываясь на обучении на исторических данных и поведении пользователей. Это позволяет значительно снизить затраты времени и ресурсов на тестирование, ускоряя цикл разработки и позволяя командам сосредоточиться на других важных задачах.

Еще одним важным преимуществом является способность нейронных сетей распознавать и анализировать визуальные элементы интерфейса. Они могут быть обучены для распознавания различных компонентов UI, таких как кнопки, текстовые поля, иконки и другие элементы. Это дает возможность автоматически выявлять визуальные дефекты или отклонения от заданного дизайна, которые могут возникнуть в результате изменений

в коде или дизайне. Такая возможность особенно полезна в условиях, когда интерфейсы часто обновляются или изменяются, что делает традиционные методы тестирования менее эффективными.

Кроме того, нейронные сети способны моделировать поведение пользователей, что открывает новые возможности для предсказания и предотвращения ошибок на ранних стадиях разработки. Они могут анализировать данные о реальных взаимодействиях пользователей с интерфейсом и выявлять элементы, вызывающие затруднения. Это позволяет улучшить пользовательский опыт путем оптимизации интерфейса еще до того, как продукт будет выпущен на рынок. Таким образом, использование нейронных сетей не только повышает эффективность тестирования, но и способствует созданию более удобного и интуитивно понятного пользовательского интерфейса.

Несмотря на очевидные преимущества, использование нейронных сетей для автоматизации UI-тестирования сопряжено с рядом вызовов. Во-первых, для эффективного обучения нейросетей требуются большие объемы данных, что может быть проблемой, особенно на ранних стадиях разработки продукта. Недостаток данных может привести к недостаточной обученности модели, что, в свою очередь, негативно скажется на точности и надежности тестирования. Для преодоления данной проблемы применяются методы генерации синтетических данных, расширения данных (data augmentation), а также техники активного обучения, которые позволяют оптимизировать процесс сбора данных и обеспечить модель достаточным количеством информации.

Во-вторых, процесс настройки и обучения нейронных сетей является сложным и требует глубоких знаний в области машинного обучения и анализа данных. Создание и обучение нейросетей требуют специализированных навыков, что может увеличить затраты на разработку и потребовать привлечения дополнительных специалистов. Кроме того, сам процесс обучения может занять значительное время, что может замедлить процесс разработки продукта, особенно если модель требует частых доработок. Для минимизации данного проблемного аспекта, возможно применение стратегии предобученных моделей и автоматизации некоторых этапов разработки нейронных сетей, например, через AutoML, что позволяет снизить требуемый уровень компетенций и ускорить процесс создания модели.

Еще одной сложностью является интерпретируемость результатов, полученных с использованием нейронных сетей. В отличие от традиционных методов тестирования, где результаты легко объяснимы и предсказуемы, нейронные сети часто действуют как "черный ящик", что затрудняет понимание причин выявленных дефектов или аномалий. Это может осложнить процесс принятия решений и требует разработки дополнительных методов для интерпретации результатов, что также добавляет сложности в процесс тестирования. Для решения этой проблемы применяются методы интерпретации и визуализации работы нейросетей, такие как Grad-CAM, LIME или SHAP, которые позволяют получить представление о том, какие части данных оказали наибольшее влияние на решение модели. Эти методы способствуют улучшению прозрачности и доверия к результатам, что важно для успешного внедрения ИИ в процессы тестирования пользовательского интерфейса.

Таким образом, несмотря на значительные преимущества, использование нейронных сетей для автоматизации UI-тестирования требует тщательного планирования и учета возможных вызовов. Для успешного внедрения таких технологий необходимо не только наличие достаточных данных и ресурсов, но и глубокое понимание принципов работы нейронных сетей, а также способность интерпретировать и использовать полученные результаты для улучшения качества продукта [8]

**Практическое применение и перспективы использования нейронных сетей для автоматизации тестирования.** Интеграция моделей нейронных сетей в существующие инструменты автоматизированного тестирования пользовательского интерфейса (UI) представляет собой современный подход к повышению эффективности и точности

тестирования. Однако для успешной интеграции нейронных сетей в инструменты автоматизированного тестирования UI важно учитывать вопросы производительности. В силу того, что модели глубокого обучения требуют значительных вычислительных ресурсов, что может повлиять на скорость выполнения тестов. В связи с этим необходимо оптимизировать модели, использовать распределенные вычисления или облачные решения для обеспечения необходимой производительности [9]. Ниже в таблице 2 будет проведено сравнение нейронных сетей с традиционными методами тестирования в тестировании пользовательского интерфейса.

**Таблица 2. Сравнение нейронных сетей с традиционными методами тестирования в тестировании пользовательского интерфейса [10] /**

**Table 2. Comparison of neural networks with traditional testing methods in user interface testing [10]**

| Критерий              | Нейронные сети                                                             | Традиционные методы тестирования                                   |
|-----------------------|----------------------------------------------------------------------------|--------------------------------------------------------------------|
| Точность предсказаний | Высокая, особенно в сложных задачах с большим объемом данных               | Средняя, зависит от человеческого фактора                          |
| Время выполнения      | Зависит от сложности модели, но обычно выше, чем у традиционных методов    | Быстрое, но требует больше итераций                                |
| Затраты               | Высокие начальные затраты на обучение и инфраструктуру                     | Низкие или средние, в зависимости от сложности тестов              |
| Сложность анализа     | Высокая, требуются специальные знания для интерпретации результатов        | Средняя, результаты обычно понятны и интерпретируемы               |
| Гибкость              | Высокая, возможно адаптировать к разным задачам и данным                   | Ограниченная, каждый тест разработан для определенной задачи       |
| Автоматизация         | Полная, после обучения нейронной сети возможно автоматическое тестирование | Частичная, требуется участие человека на каждом этапе              |
| Масштабируемость      | Высокая, может работать с большими объемами данных и различными сценариями | Ограниченная, требует ручной настройки для каждого нового сценария |
| Проблемы и риски      | Возможность переобучения, потребность в больших объемах данных             | Человеческие ошибки, ограниченная применимость                     |

Если же говорить о будущем применения нейронных сетей в тестировании пользовательского интерфейса, то оно представляет собой одну из наиболее перспективных областей развития технологий. Нейронные сети могут быть применены для распознавания и анализа различных паттернов взаимодействия пользователей с интерфейсами. С их помощью можно автоматически выявлять аномалии, определять слабые места в дизайне, а также предлагать возможные улучшения на основе анализа большого количества данных. Это особенно важно в условиях увеличивающейся сложности интерфейсов и разнообразия пользовательских сценариев. Альтернативно, нейросети могут быть использованы для упрощения процесса принятия решений. Нейросети могут использоваться, чтобы дать разработчикам или тестировщикам инсайты относительно надежности продукта, влиять на разработку тестов и оценивать возможные ветви погрешностей по сравнению с отображенными решениями [11].

**Заключение.** Применение нейронных сетей для автоматизации тестирования пользовательского интерфейса открывает новые горизонты в области обеспечения качества программного обеспечения. Внедрение этих технологий позволяет существенно

ускорить процессы тестирования, повысить их точность и адаптивность к изменениям. Однако для полной реализации потенциала нейронных сетей требуется дальнейшее развитие методик и инструментов, а также преодоление существующих технических и организационных барьеров. В заключение можно отметить, что интеграция нейросетевых технологий в процессы разработки станет ключевым фактором повышения конкурентоспособности и качества цифровых продуктов в будущем.

#### ЛИТЕРАТУРА

1. Cherniltsev A. G. Application of a multilayer neural network in user interface update tasks // Cybernetics and Programming. 2018. Vol. 3. P. 38–47.
2. Faustova K. I. Neural networks: application today and development prospects // The Territory of Science. 2017. Vol. 4. P. 83–87.
3. Ivanko A. F., Ivanko M. A., Kolesnikova O. D. Information neural networks // Scientific Review. Technical Sciences. 2019. Vol. 4. P. 11–16.
4. Sushchenya R. V., Kokaev A. E. Neural networks and their classification. The main types of neural networks // Bulletin of Science. 2023. Vol. 8. No. 65. P. 186–189.
5. Trifonov K. V. Comparison of convolutional and recurrent architectures of neural networks in solving the problem of text tonality analysis // Young Researcher of Don. 2024. Vol. 2. P. 41–44.
6. AI in Test Automation: Here's How It Works. [Электронный ресурс]. URL: <https://thectoclub.com/ai-ml/ai-test-automation/> (дата обращения: 08.09.2024).
7. 10 Top UI Test Automation Tools in 2024: Empowering Efficient and Robust Testing Strategies. [Электронный ресурс]. URL: <https://www.towardsanalytic.com/10-top-ui-test-automation-tools-in-2024-empowering-efficient-and-robust-testing-strategies/> (дата обращения: 09.08.2024).
8. Plotnikov Ya. M., Novikov D. V. The effectiveness of the use of neural networks in game testing. Instrumentation and automation: maintenance and repair. 2021. [Электронный ресурс] URL: <https://panor.ru/articles/effektivnost-primeneniya-neyronnykh-setey-v-testirovanii-igr/55802.html> (дата обращения: 08.09.2024).
9. Устименко Л. Р., Билека Т. О., Сафонов И. А. Применение машинного обучения в автоматизации тестирования ПО // Научные исследования студентов и учащихся: сборник статей XI Международной научно-практической конференции. Пенза: Наука и Просвещение, 2024. С. 34–38.
10. Yurkin V. A., Saradgishvili S. E., Voinov N. V., Molodyakov S. A. Applying neural network to assess application user experience // 2023 IV International Conference on Neural Networks and Neurotechnologies (NeuroNT). Saint Petersburg: IEEE, 2023. P. 39–42.
11. Mokshanov M. V. The use of artificial intelligence in data analysis: an overview of the current state and future directions // Universum: Technical Sciences. 2024. Vol. 1. No. 5. P. 40–48.

#### REFERENCES

1. Cherniltsev AG. Application of a multilayer neural network in user interface update tasks. Cybernetics and Programming. 2018;3:38-47.
2. Faustova KI. Neural networks: application today and development prospects. The Territory of Science. 2017;4:83-87.
3. Ivanko AF, Ivanko MA, Kolesnikova OD. Information neural networks. Scientific Review. Technical Sciences. 2019;4:11-16.
4. Sushchenya RV, Kokaev AE. Neural networks and their classification. The main types of neural networks. Bulletin of Science. 2023;8(65):186-189.
5. Trifonov KV. Comparison of convolutional and recurrent architectures of neural networks in solving the problem of text tonality analysis. Young Researcher of Don. 2024;2:41-44.
6. AI in Test Automation: Here's How It Works. Available from: <https://thectoclub.com/ai-ml/ai-test-automation/> [Accessed 8 September 2024].
7. 10 Top UI Test Automation Tools in 2024: Empowering Efficient and Robust Testing Strategies. Available from: <https://www.towardsanalytic.com/10-top-ui-test-automation-tools-in-2024-empowering-efficient-and-robust-testing-strategies/> [Accessed 9 August 2024].
8. Plotnikov YM, Novikov DV. The effectiveness of the use of neural networks in game testing. Instrumentation and automation: maintenance and repair. 2021. Available from:

<https://panor.ru/articles/effektivnost-primeneniya-neyronnykh-setey-v-testirovanii-igr/55802.html>

[Accessed 8 September 2024].

9. Ustimenko LR, Bileka TO, Safonov IA. Application of machine learning in software testing automation // Scientific research of students and pupils: collection of articles of the XI International scientific and practical conference. Penza: Science and Education, 2024. P. 34-38. (In Russ.).
10. Yurkin VA, Saradgishvili SE, Voinov NV, Molodyakov SA. Applying neural network to assess application user experience // 2023 IV International Conference on Neural Networks and Neurotechnologies (NeuroNT). Saint Petersburg: IEEE, 2023. P. 39-42.
11. Mokshanov MV. The use of artificial intelligence in data analysis: an overview of the current state and future directions // Universum: Technical Sciences. 2024;1(5):40-48.

#### ИНФОРМАЦИЯ ОБ АВТОРЕ

**Игорь Георгиевич Волынец** – старший специалист по тестированию ПО. Юзтех Бел, [ivolynets@usotech.ru](mailto:ivolynets@usotech.ru)

**Конфликт интересов:** автор заявляет об отсутствии конфликта интересов.

Статья поступила в редакцию: 01.04.2025;  
одобрена после рецензирования: 12.05.2025;  
принята к публикации: 01.06.2025.

#### INFORMATION ABOUT THE AUTHOR

**Igor G. Volynets** – Senior QA Automation Engineer. Usetech Bel, [ivolynets@usotech.ru](mailto:ivolynets@usotech.ru)

**Conflict of interest:** the author declares no conflicts of interests.

The article was submitted: 01.04.2025;  
approved after reviewing: 12.05.2025;  
accepted for publication: 01.06.2025.